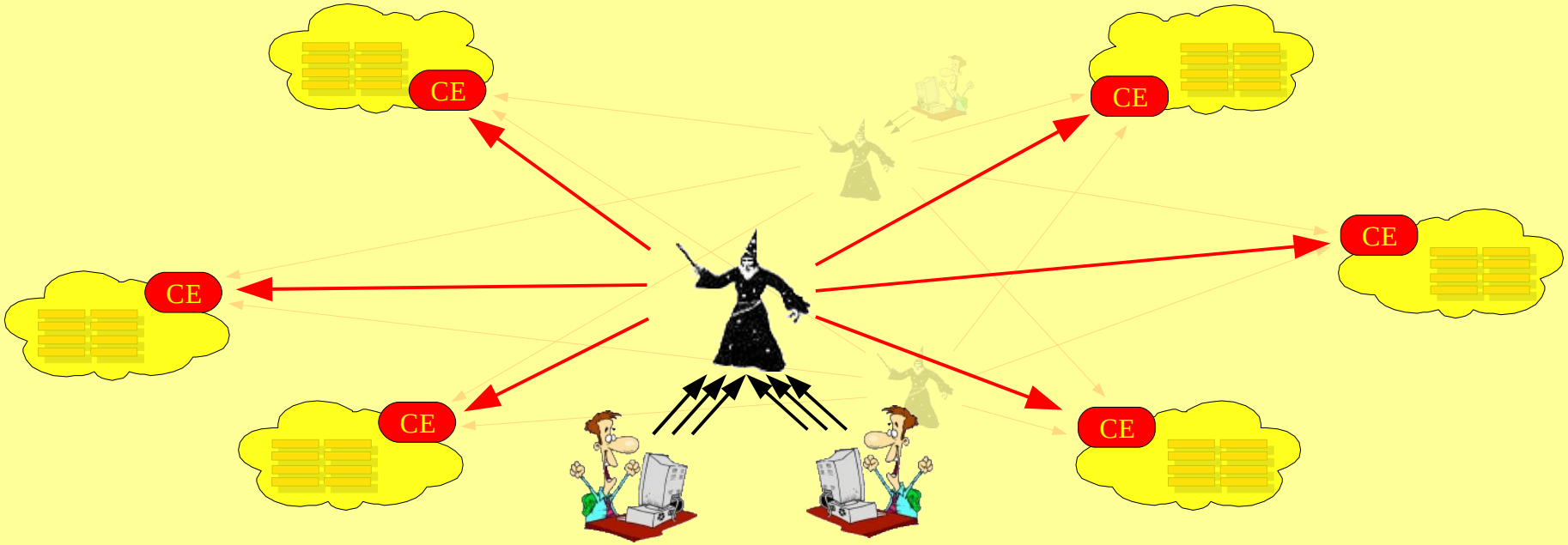
Workload Management Systems Evaluation and Integration



by Igor Sfiligoi & Burt Holzman

Scope / Timeline / Deliverables

- There exist a proliferation of workload management systems (WMS) available across the worldwide LHC Computing Grid and Open Science Grid
- US CMS requires a quantitative evaluation of the WMS solutions on the market prior to integration with experiment-specific production and analysis front-ends

✓ • 4th quarter 2006: Build test infrastructure, convert CDF glidein work to generic glideinWMS

✓ • 1st quarter 2007: Collect data, evaluate products

- 2nd quarter 2007: Integrate chosen solution(s) into production ("ProdAgent")
  - Prototype 1.0 before CSA07
- 3rd quarter 2007: Integration into analysis server ("CRAB Server")
  - Version 2.0 in stable production use post-CSA07
- 4th quarter 2007 and beyond: maintenance

Effort Profile

- 2006-2007: evaluation, development, integration
  - 1 FTE (50% Sfiligoi, 50% Holzman)
- 2008-20xx: maintenance, support, operations
  - .25 FTE (25% Sfiligoi / others)

Risks

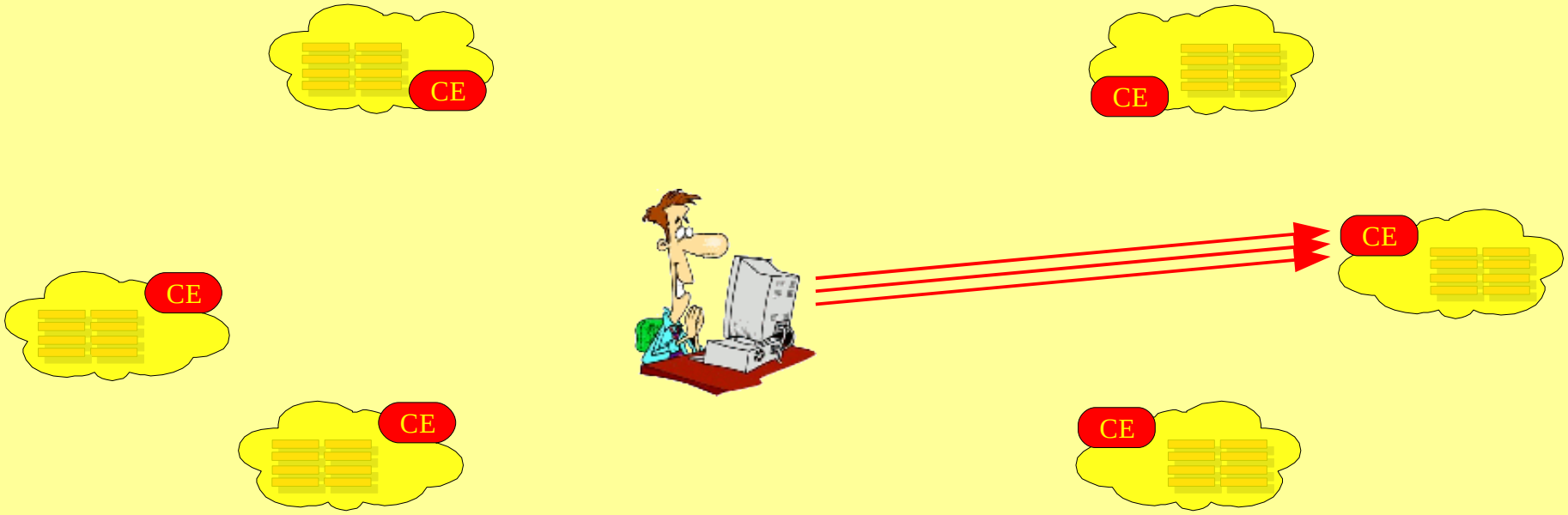
- Incomplete or qualitative evaluation may lead to choice of poorly-performing WMS, affecting scalability (hardware costs), usability (support effort), overall success rates (efficiency)
- Lack of integration with production and analysis infrastructure may also lead to issues with grid efficiency and the inability to efficiently use experiment-funded resources (and to opportunistically use the grid!)
- Schedule slips are tolerable:
  - Front-end API for production and analyses is nearly fixed, but back-end integration to WMS is flexible and can happen at any time
  - LHC has slow ramp-up for the 1st year or so

What did we test

- Scalability and reliability
  - in a single user environment
  - using 4 Grid sites, ghost pool overlaid w/production pool (Caltech, Fermilab, Madison, UCSD)
  - running simple sleep jobs(0.4-5h), using small I/O files
- Tested WMSes
  - Plain Condor-G (http://www.cs.wisc.edu/condor/manual/v6.9/5_3Grid_Universe.html)
  - ReSS (https://twiki.grid.iu.edu/bin/view/ResourceSelection/)
  - gLite WMS (http://glite.web.cern.ch/glite/documentation/)
  - glideinWMS (http://home.fnal.gov/~sfiligoi/glideinWMS/)

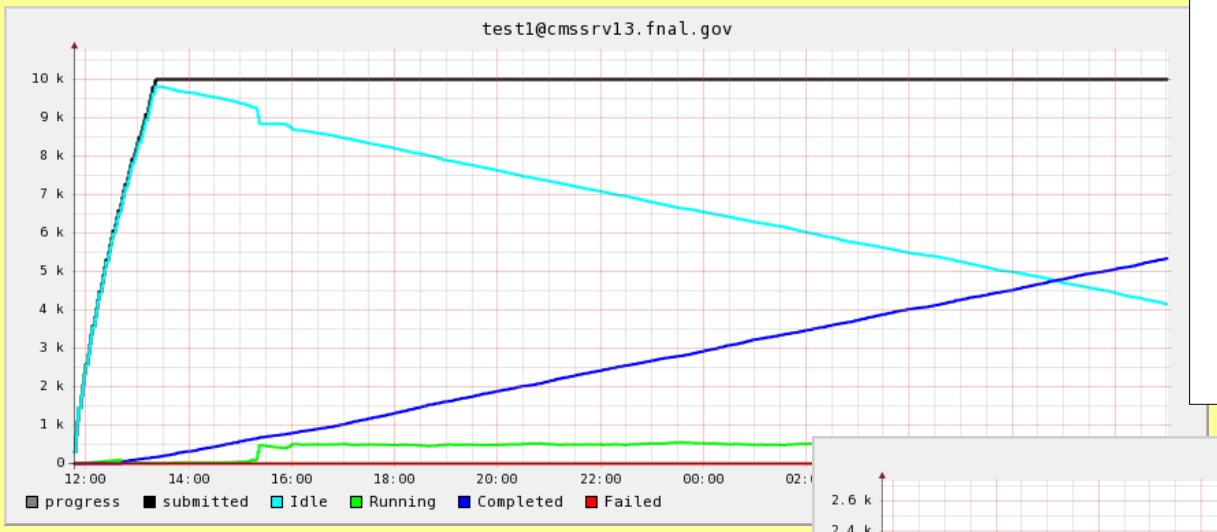
Plain Condor-G

- Manual selection of the site
  - Base test to verify CE scalability and reliability



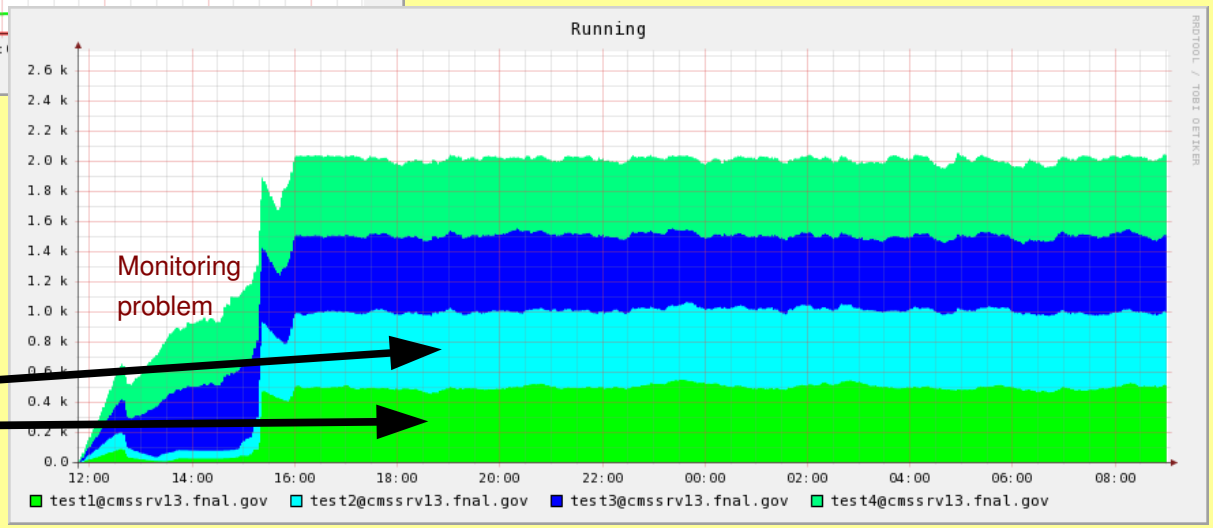
Condor-G scalability

- Scales nicely, no problems found up to 4x10k



Submitted
Idle
Running
Completed
Failed

~2k batch slots at Grid site

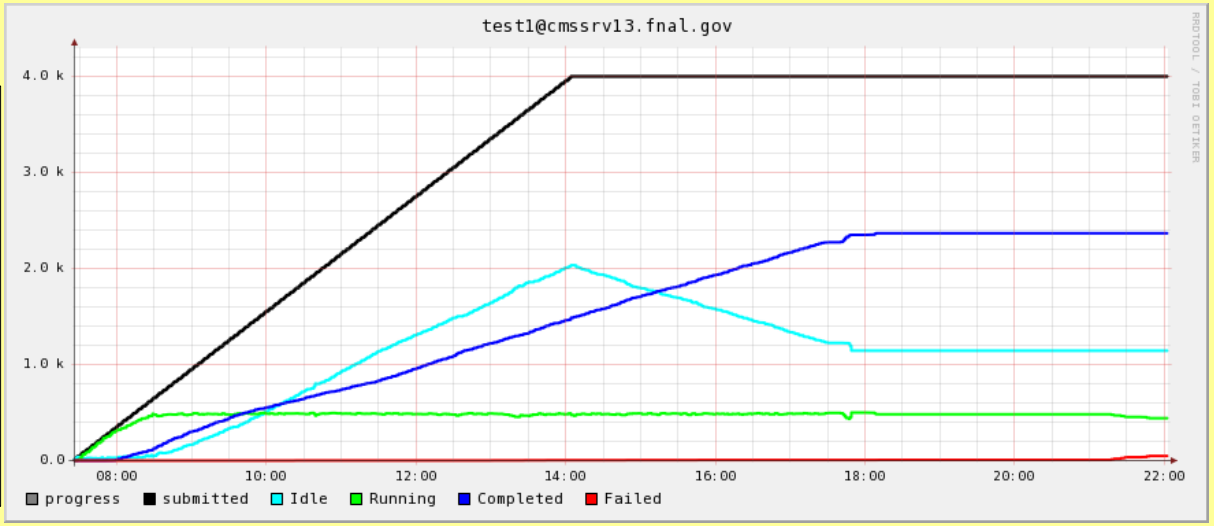


Each color is a different schedd

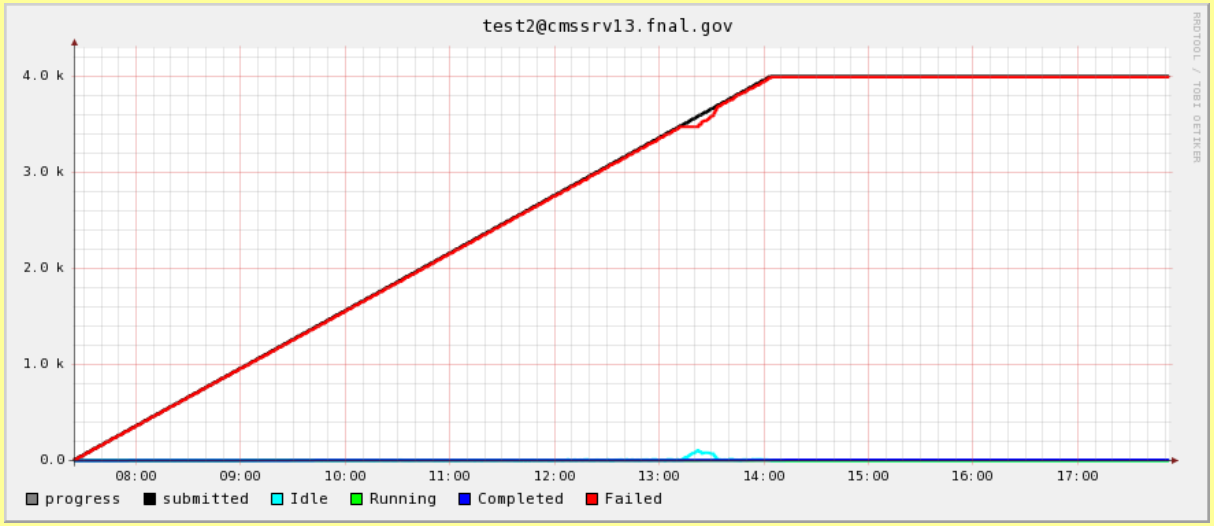
Condor-G reliability

- Works fine when Grid site stable

Submitted
Idle
Running
Completed
Failed



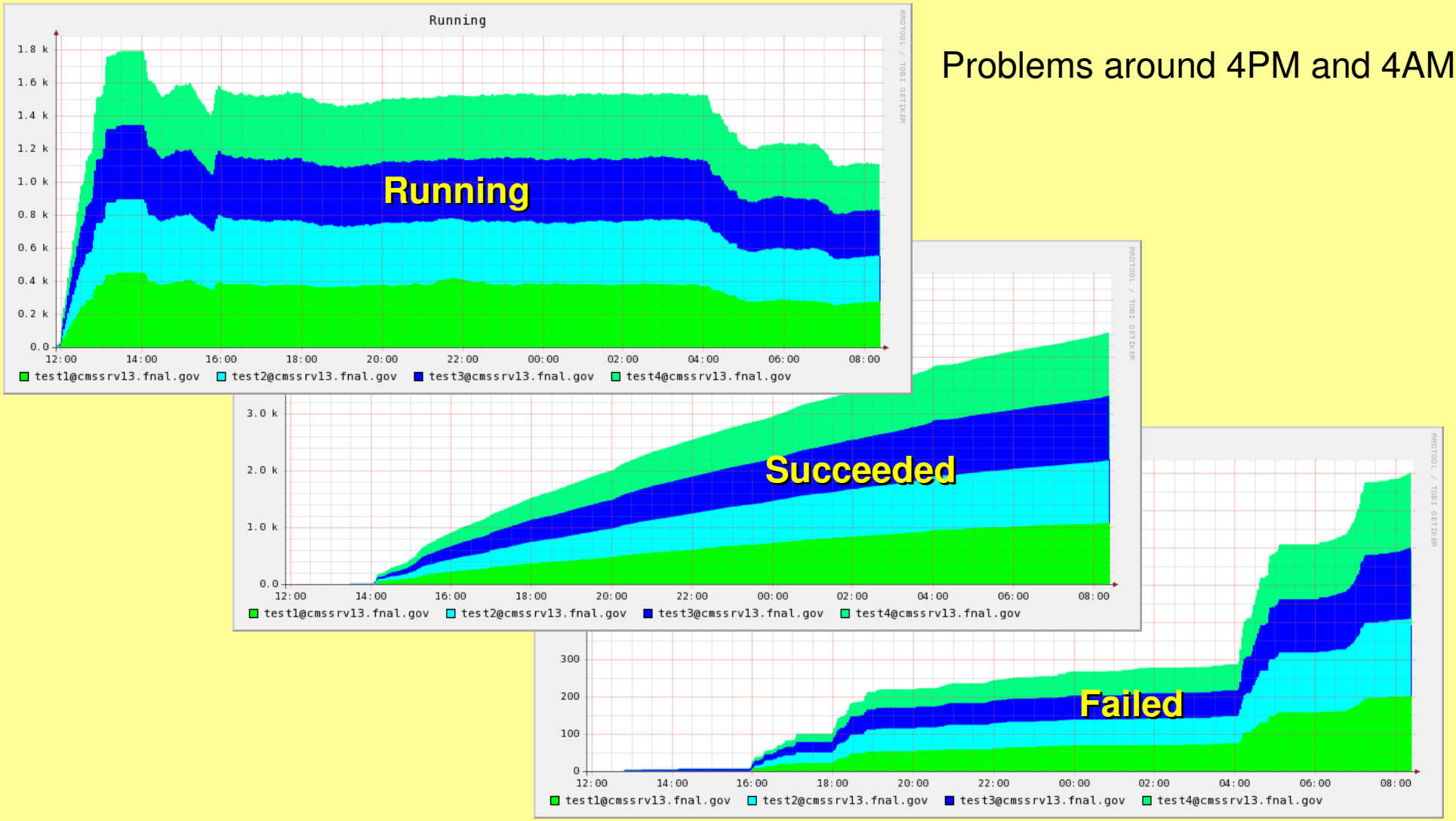
- But lots of jobs fail when Grid site misbehaves
  - Nothing that can be done on the client side



This site worked perfectly 24h ago

Condor-G reliability (2)

- Another example



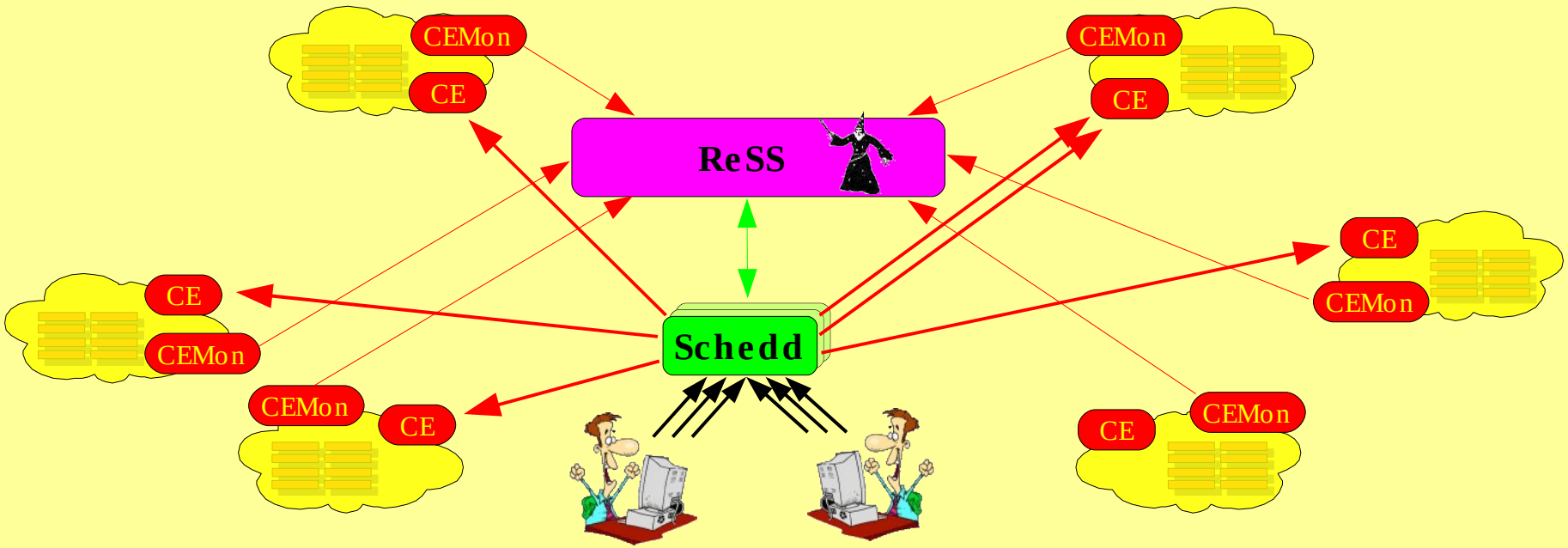
Problems around 4PM and 4AM

Condor-G reliability

- Condor-G does not handle well Grid CE crashes
  - If jobs are removed from the Grid queue before the CE comes back, Condor-G still thinks all the jobs are still there
  - If the GridMonitor process gets killed on the CE, Condor-G loses all control over the jobs that were managed by it
- Several times substantial differences between what Condor-G thinks is queued and what was actually queue have been observed

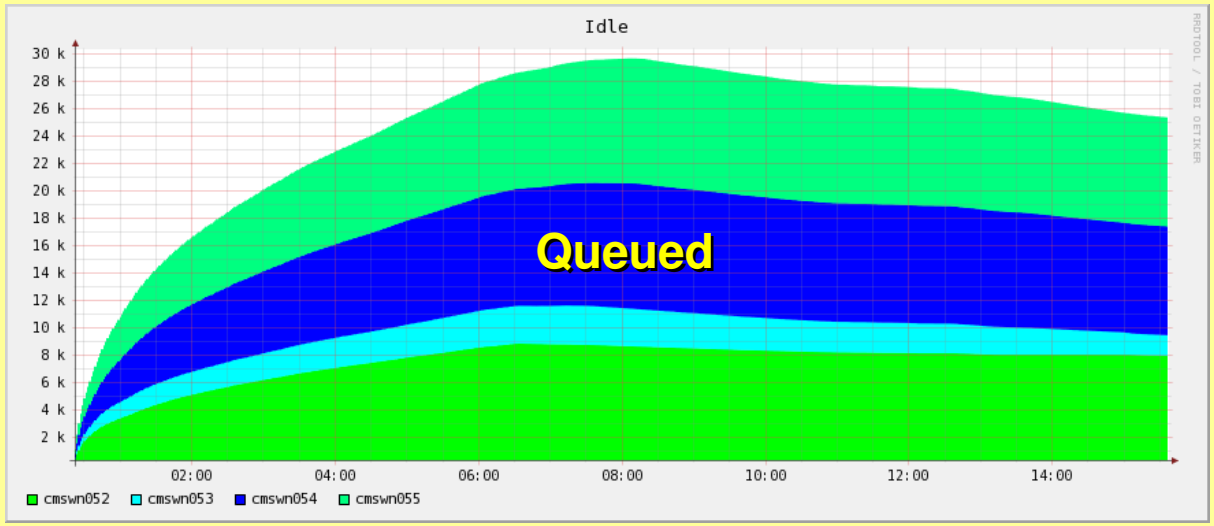
ReSS

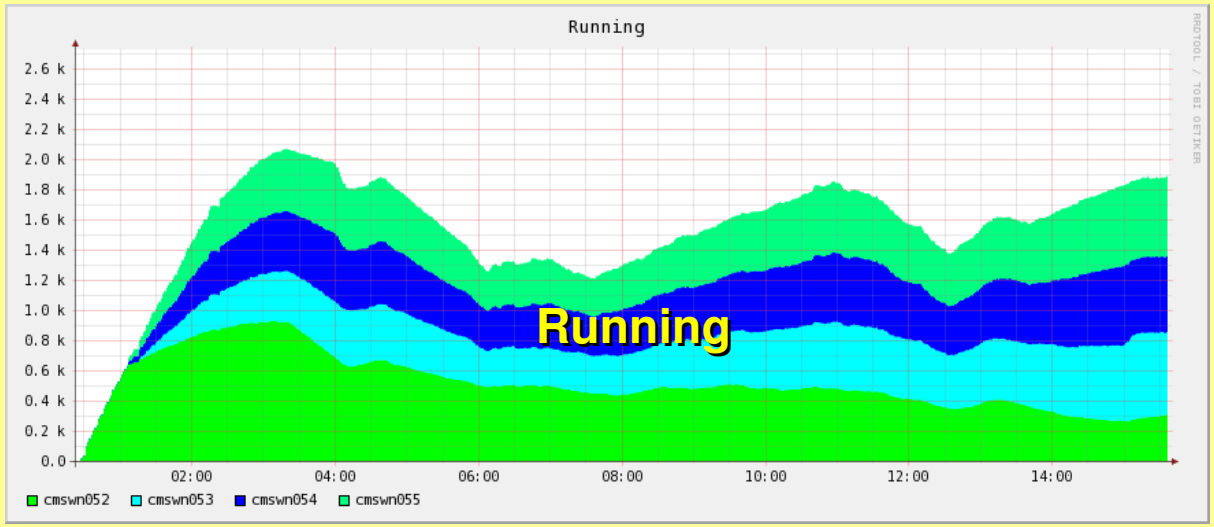
- A Condor-G based system
  - ReSS selects the Grid site for the user
  - Needs information from the Grid sites (CEMon in OSG v0.6)



ReSS scalability

- No problem up to 4x10k queued
  - Had to test on
    a single Grid pool
    (the only w/CEMon)

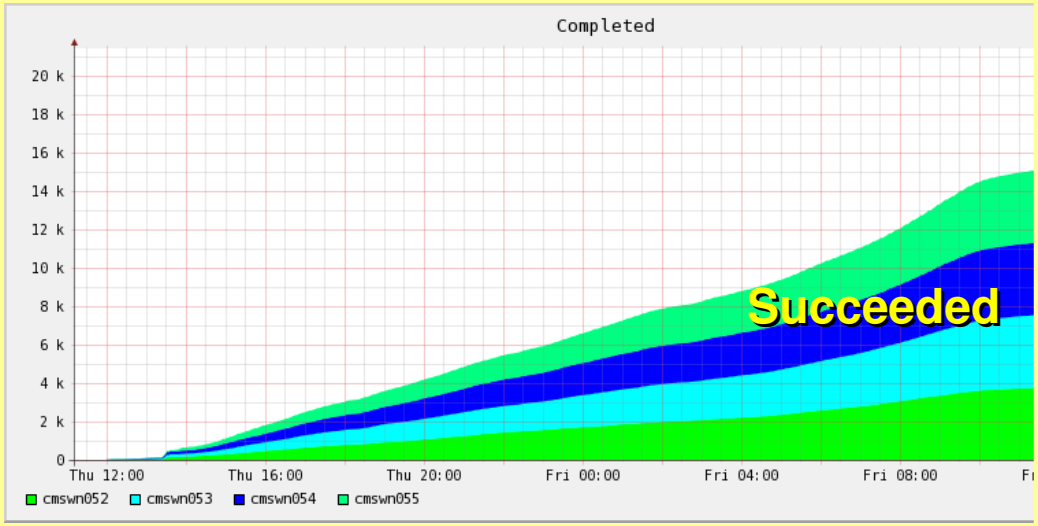


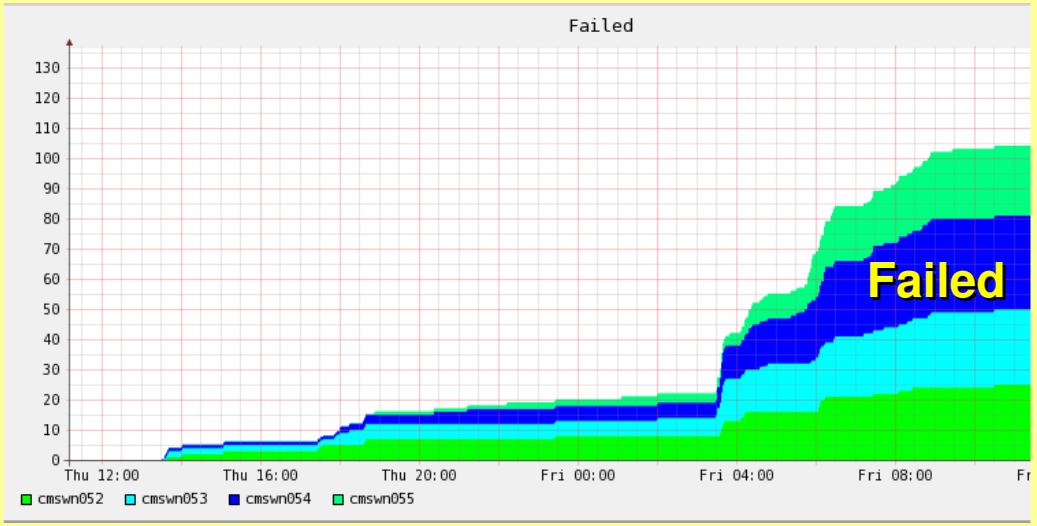


2k slots on Grid site

ReSS reliability

- Similar to Condor-G

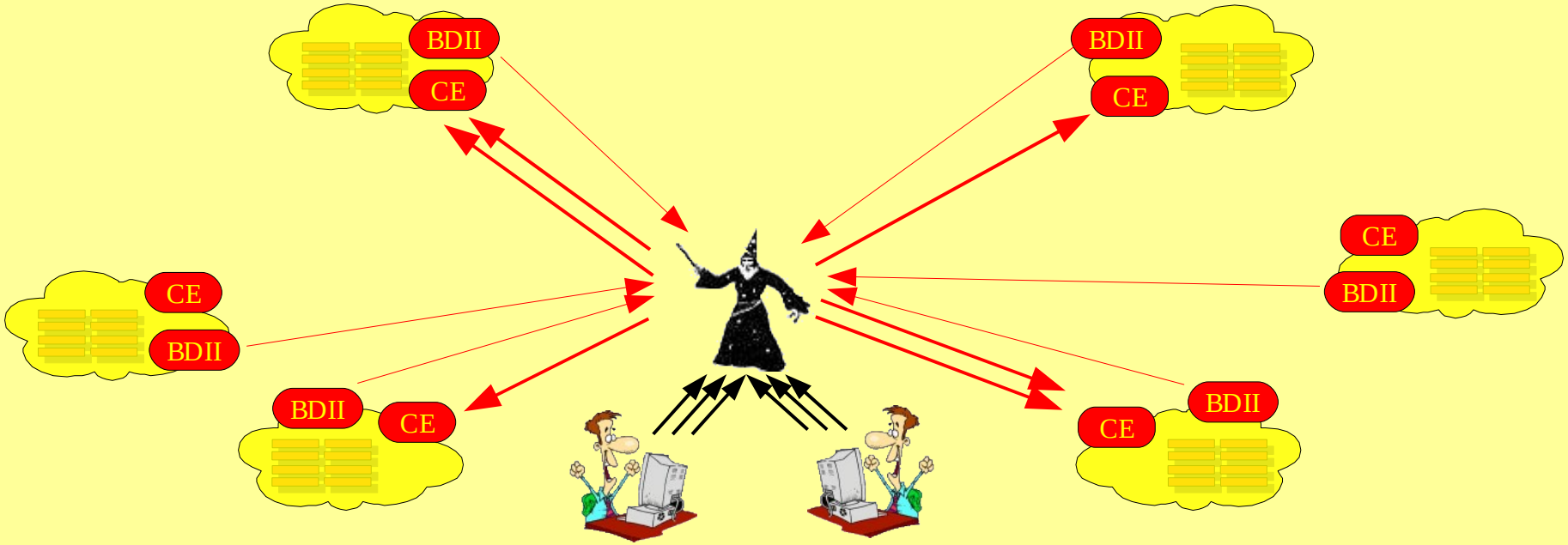




- Potentially, misconfigured CEMon can send jobs to the wrong Grid site
  - At least on paper... unfortunately, tested with just one site
- Certain failures could pot. be automatically recovered
  - Not out the box, not tested

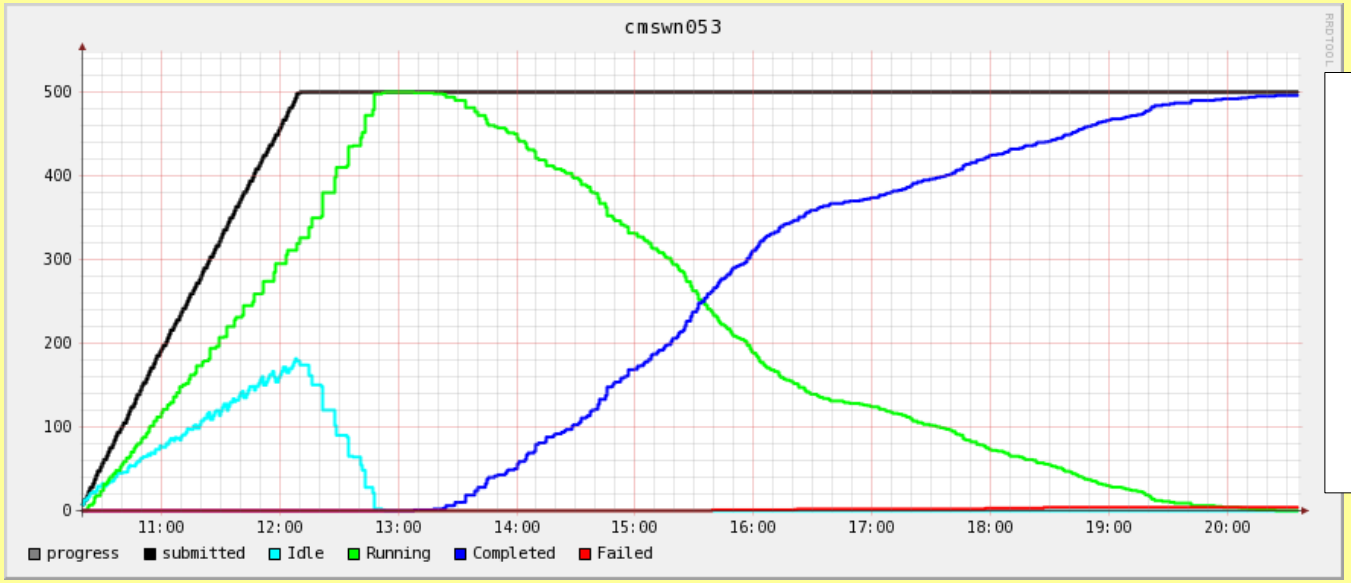
gLite WMS

- A black box solution, needs dedicated client
- Needs support from Grid sites
  - BDII for site information (OSG, wLCG)
  - gLite tools for job execution (~~OSG~~, wLCG)



gLite WMS scalability (1)

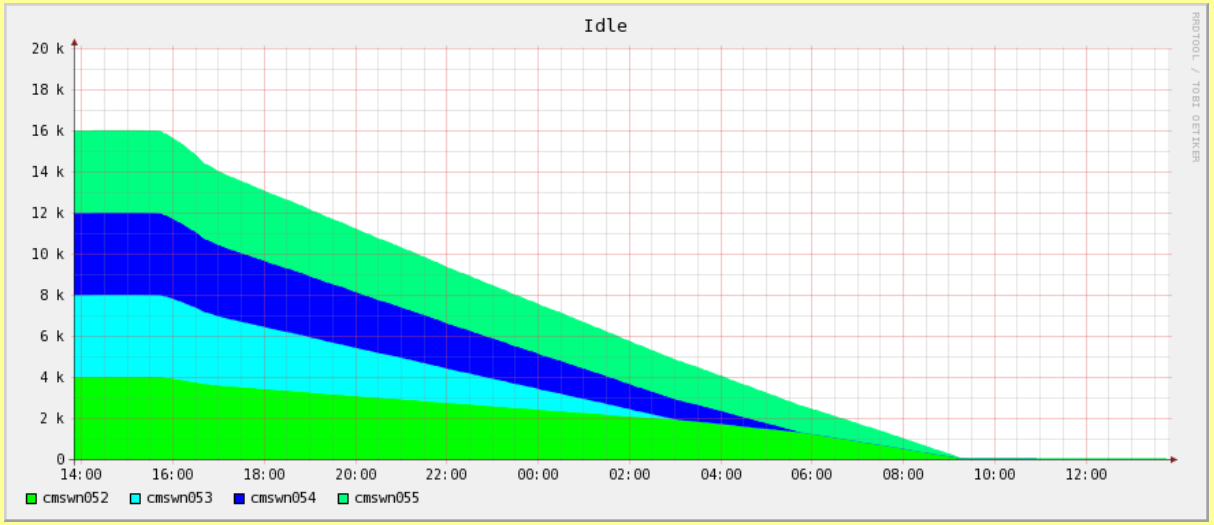
- The normal submission impractical past 4x500
  - Took 2 hours to submit (4x10k would take at least 40h!)

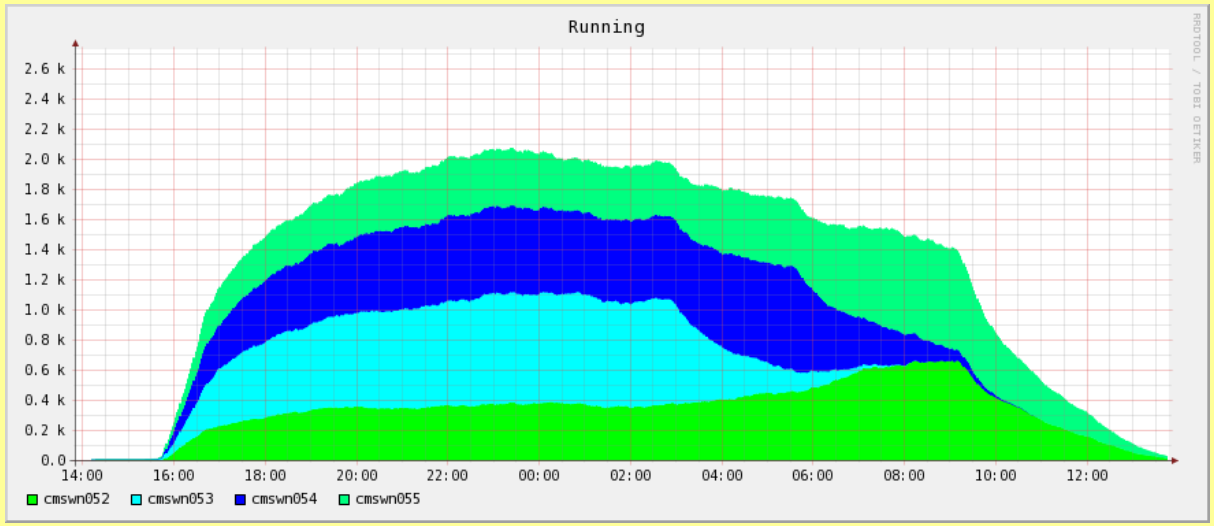


Submitted
Idle
Running
Completed
Failed

gLite WMS scalability (2)

- Bulk mode much faster: 4x4k submitted in 20mins





2 Grid sites

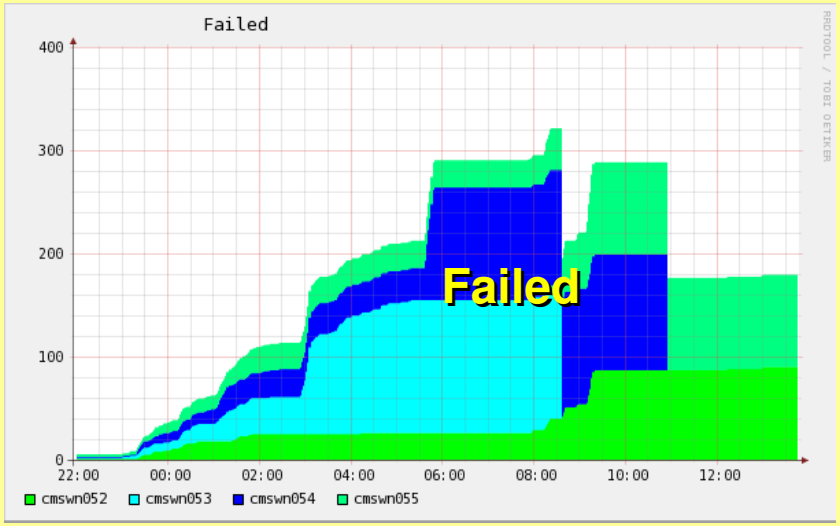
~2.5k Grid slot

gLite WMS scalability (3)

- The system was quite loaded at 4x4k
- Were not able to run 4x10k
  - All four clients reported errors on submission
- Similarly, 15x2k was disappointing
  - 12 out of 15 clients reported errors on submission (and each client tries 3 times)

gLite WMS reliability

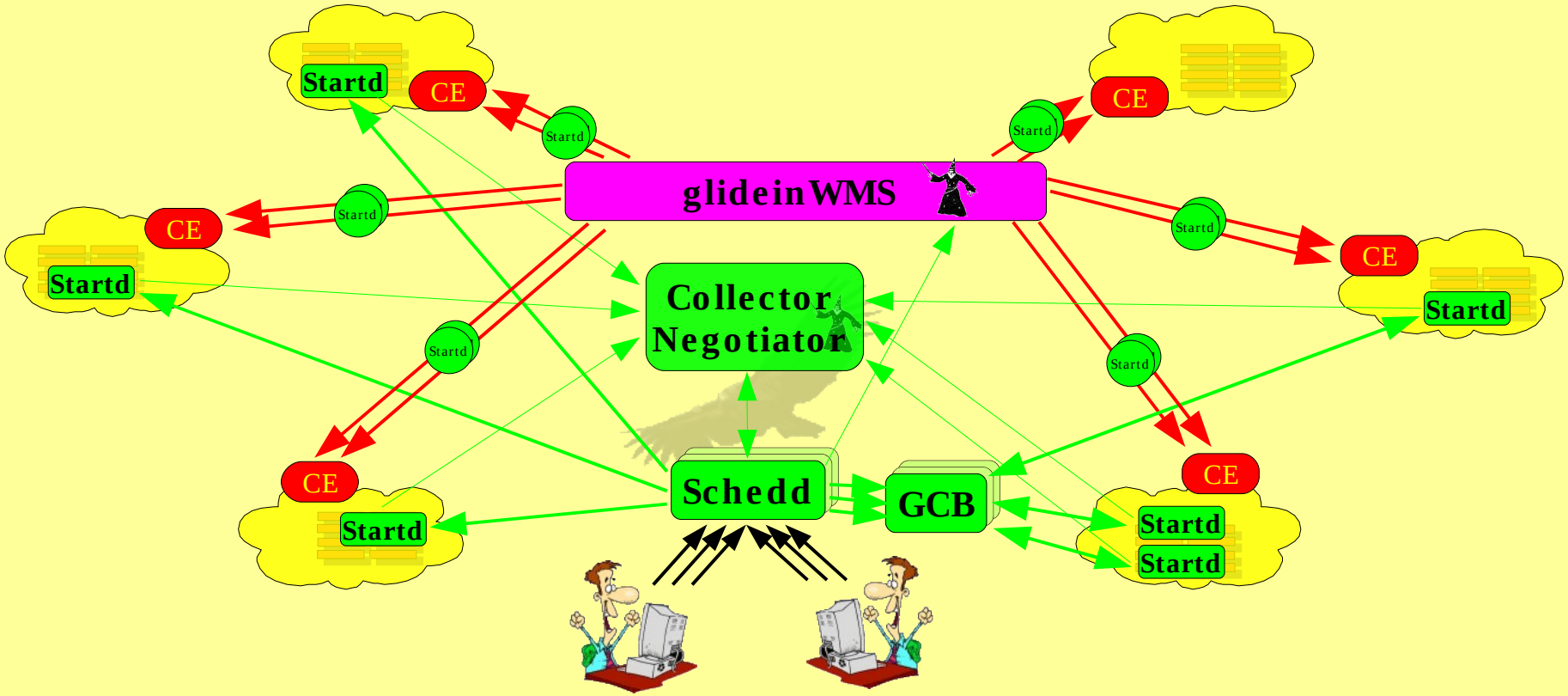
- Internally uses Condor-G, so most problems the same
  - But it does retry a job several times if submission fails
    - Still several jobs failed at every try



- Potentially, missconfigured BDII can send jobs to the wrong Grid site
  - At least on paper... did not happen during the test

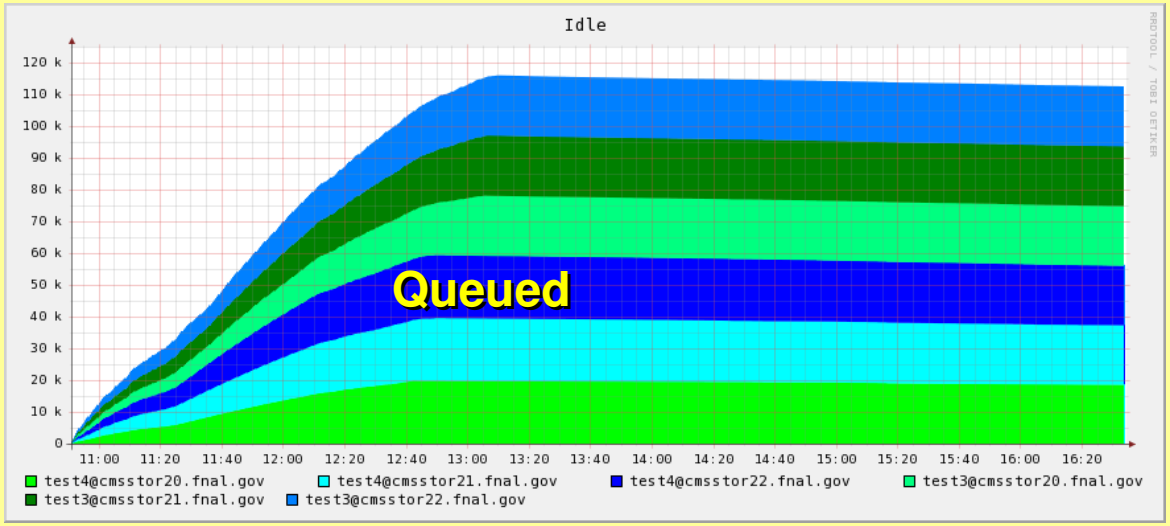
glideinWMS

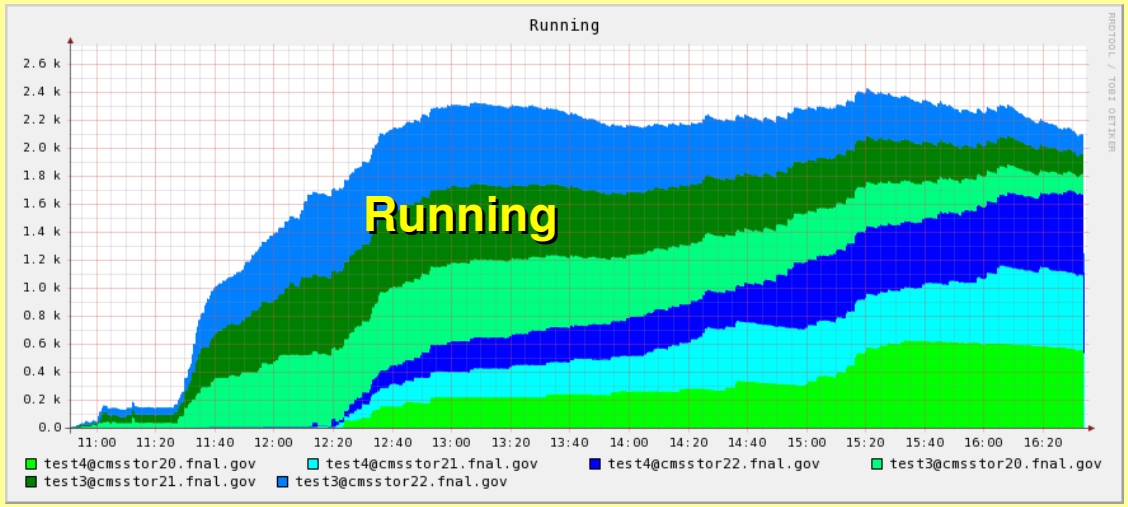
- Essentially a standard Condor pool, with startds started in a dynamic way



glideinWMS scalability

- Tested up to 6x20k jobs without finding a problem

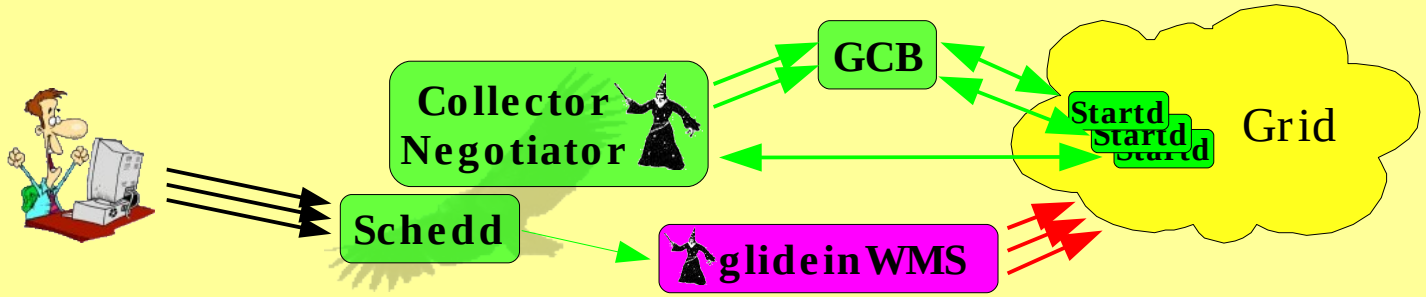




Running over
3 Grid sites

Condor scalability

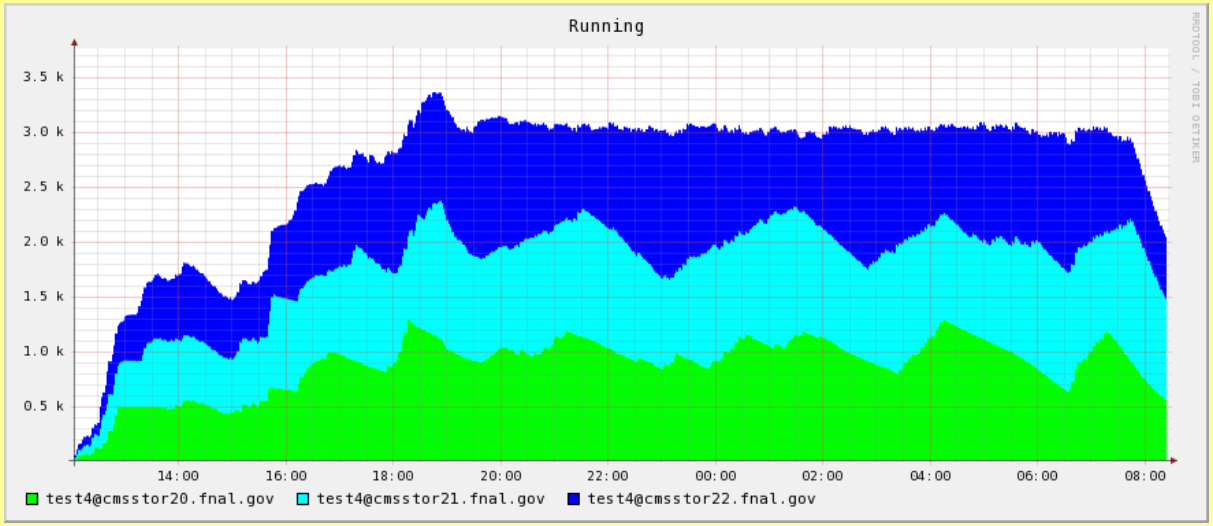
- glideinWMS just a small layer on top of Condor
  - Condor does most of the work



- Tested both Condor v6.8.x and v6.9.x branches
  - Only the latest releases of both branches scale reasonably well in the WAN environment
  - Most tests done with pre-releases, after Condor team fixed (most) observed bugs

Condor Collector scalability

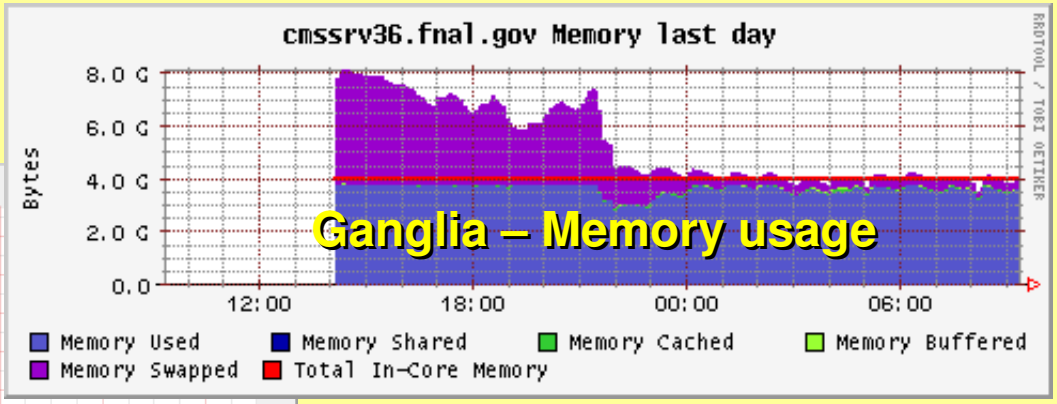
- Collector found scalable to at least 6k VMs
  - Collector was quite loaded, but jobs ran fine
  - Did not test higher, for lack of enough Grid cycles

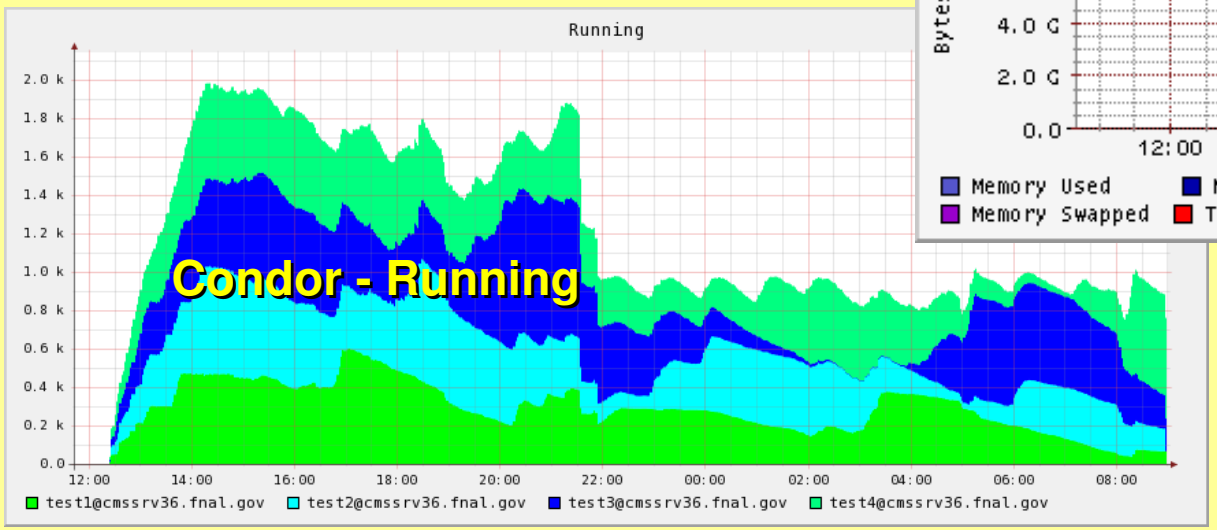


Only half VMs used by jobs in this setup

Condor Schedd scalability

- The main scalability issue found was memory consumption
  - 4M x running job!
  - Need to use multiple nodes





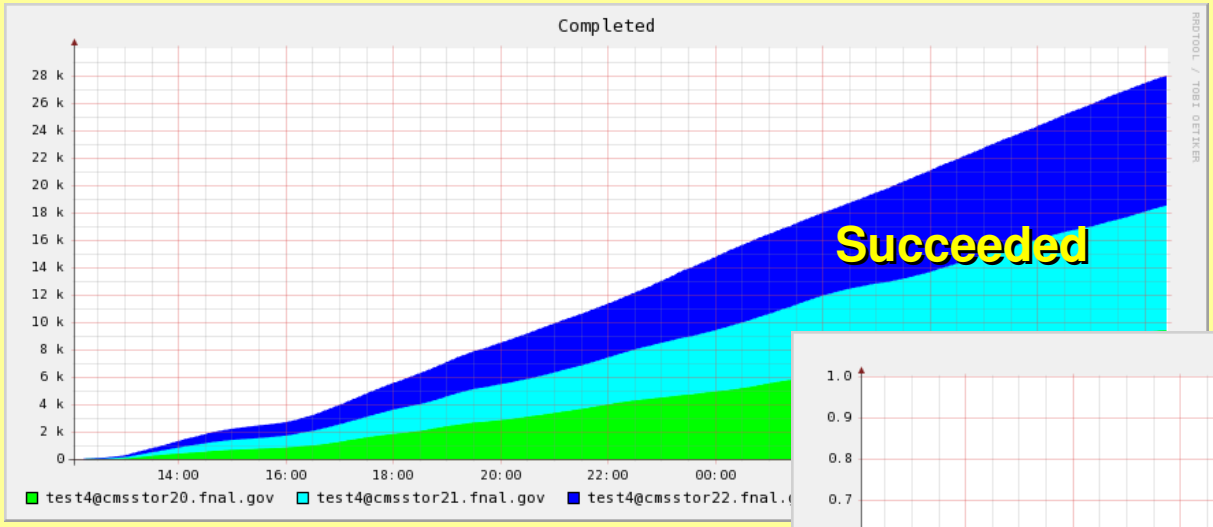
- Probably a condor bug – waiting for new dev. release
  - Regular Condor pools in OSG use less than 1M x running job

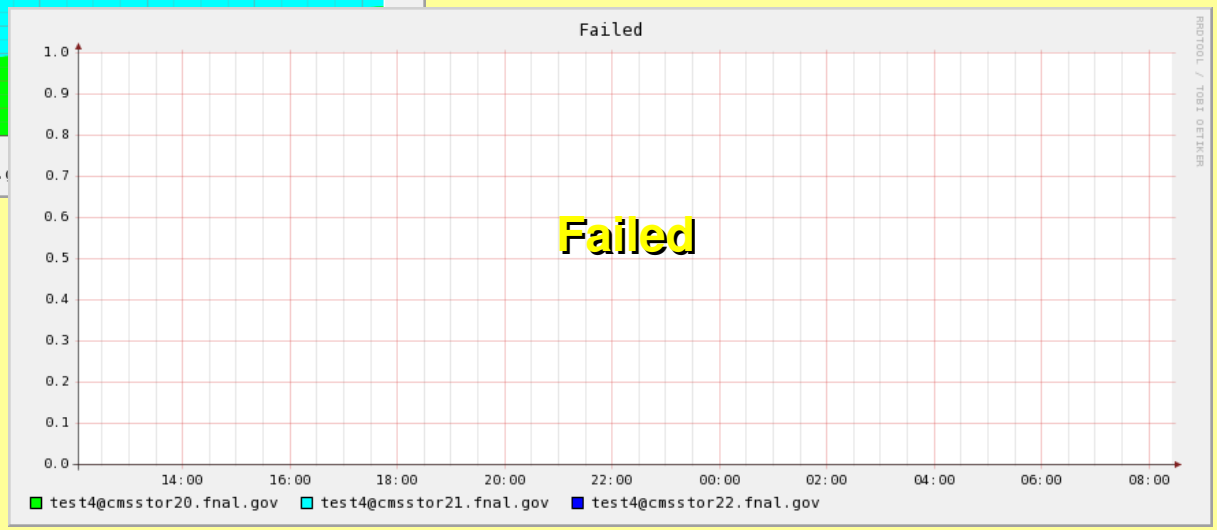
Condor GCB scalability

- Tested up to ~1500 glideins (3k VMs) per GCB
  - up to ~3k glideins with 2 GCBs
- GCB seems to scale reasonably well
  - Test jobs were running fine (with latest version)
  - However, lots of error messages seen in GCB condor logs
    - One critical problem fixed, other still under investigation
- GCB libraries sensitive to malformed packets
  - FNAL security scans occasionally crash some daemons
  - Condor team working on fixes, some in v6.9.2

glideinWMS reliability (1)

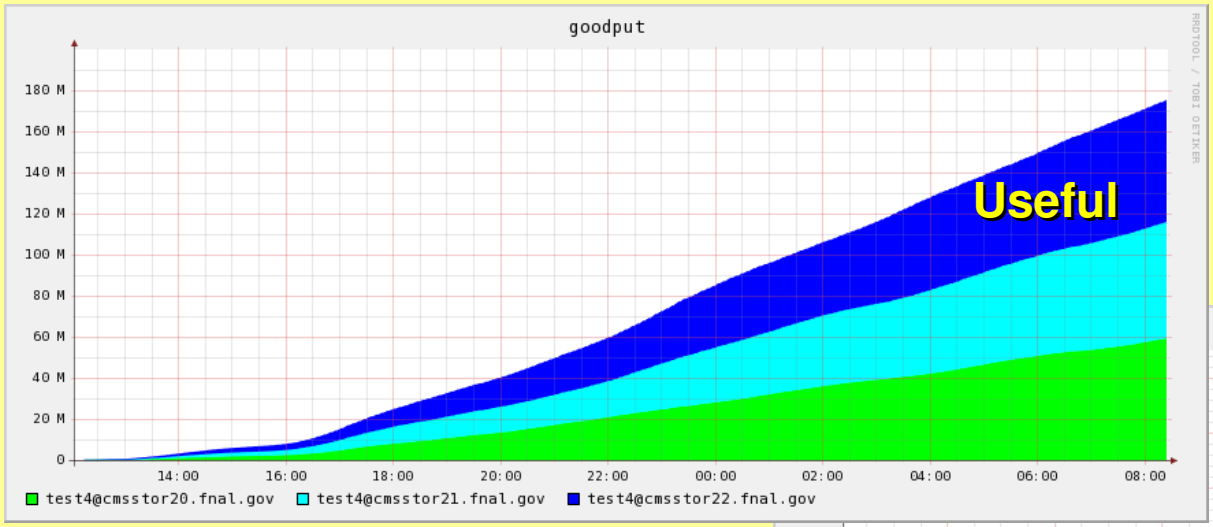
- User jobs almost never fail
  - Problematic Grid sites/nodes kill glideins not user job

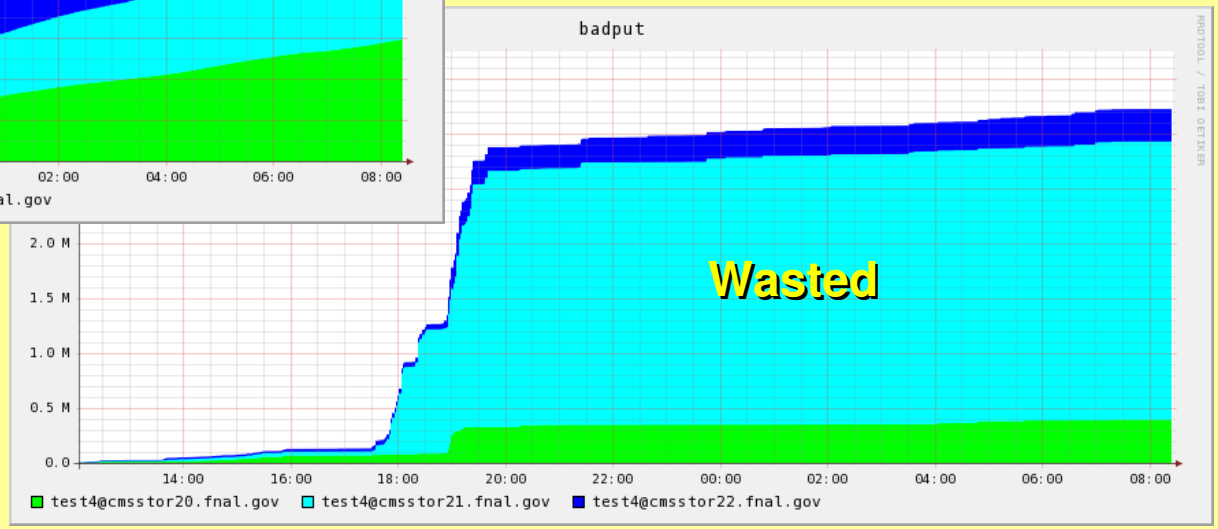




glideinWMS reliability (2)

- If glidein dies after job started, Condor will restart the user job in another glidein
  - Just wasted CPU (Checkpointing could eliminate it)





Conclusions(1)

- ReSS and glideinWMS both performed very well, gLite WMS does not scale
  - ReSS is very lightweight
    - One node can serve large number of jobs and batch slots
  - However:
    - Failures only partially handled
    - No global fair share
  - glideinWMS the most powerful
    - Virtually no job failures
    - Global fair share across Grid sites (not tested here)
  - However
    - Heavyweight, needs approx. two nodes every 2k batch slots*
    - PULL model disliked by some Grid sites
    - Needs gLExec on WN for proper security (not in OSG0.6)

Conclusions(2)

- For automated tasks involving just a few entities, ReSS may be preferable
  - Lightweight, failures can be recovered by the submitter
- For multi-user environments with varying user demands, glideinWMS is definitely the way to go if you can afford the needed hardware
  - Virtually no user job failures and real global fair share a must for the average user

Next steps

- Additional tests of ReSS and glideinWMS?
  - Bigger I/O files
  - Non-trivial applications
  - More Grid sites
  - Multiple users
- Integrate ReSS and glideinWMS into CMS MC and analysis tools
  - Performance there will be the real test